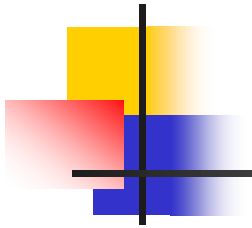
PDP1 Design Workshop

Silicon Valley
Forth Interest Group

June 25, 2016

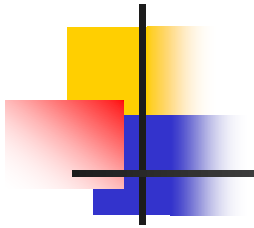
Chen-Hanson Ting



Summary

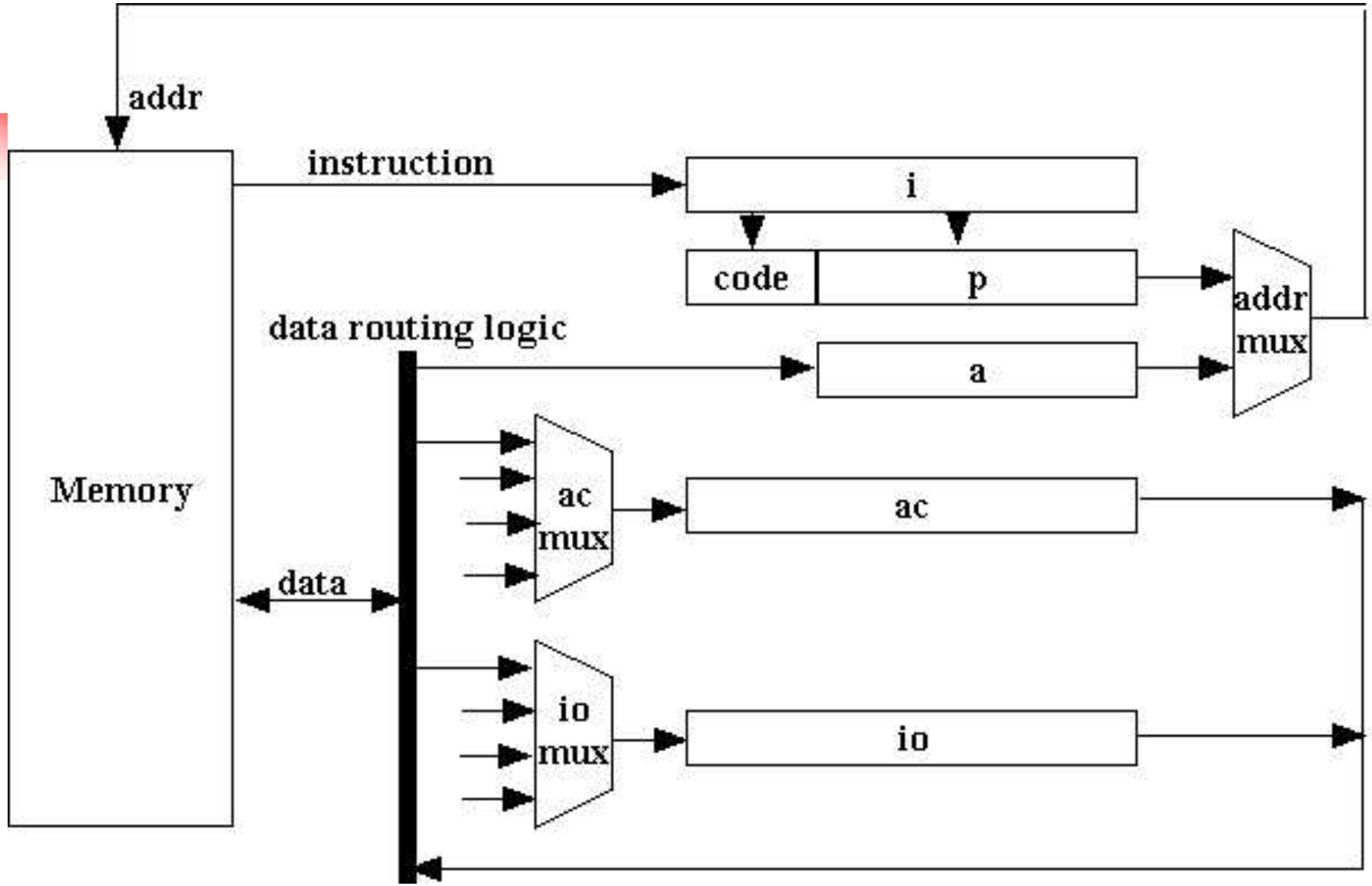
- PDP1 Architecture
- PDP1 Instruction set
- FPGA Implementation
- eForth Implementation
- Demonstration
- Discussions

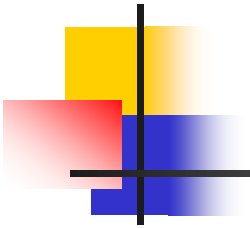

TEXT GOOD TO 474747
SCAN



PDP1 Architecture

- Von Neumann architecture
- 4096 18-Bit core memory
- AC and IO registers
- 18-Bit instructions
  - 5-bit opcode
  - 1-bit indirection
  - 12-bit address


addr
instruction
i
code
p
addr
mux
data routing logic
a
Memory
ac
mux
ac
data
io
mux
io



PDP1 Instruction set

- Memory reference instructions
  - Arithmetic instructions
  - Logic instructions
  - General instructions
- Augmented instructions
  - Literal instruction
  - Shift / rotate instructions
  - Skip instructions
  - Operate instructions

PDP1 Instruction set

Memory Reference Instructions

Op Code	I	Address

00	10 xct	20 lac	30 dip	40 add	50 sad	60 jmp	70 law
02 and	12	22	32 dio	42 sub	52 sas	62 jsp	72 ue
04 ior	14	24 dac	34 dzm	44 idx	54 mul	64 skp	74
06 xor	16 cal	26 dap	36 *	46 isp	56 div	66 sft	76 opr

ALU Instructions

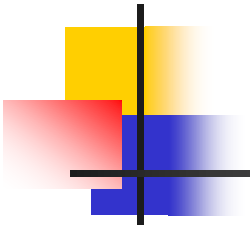
add Y	40 add
sub Y	42 subtract
mus Y	54 multiply step
dis Y	56 divide step
idx Y	44 increment
isp Y	46 increment and skip if positive
and Y	2 logical and
xor Y	6 exclusive or
ior Y	4 inclusive or

Register Instructions

lac Y	20 load accumulator
dac Y	24 deposit accumulator
dap Y	26 deposit address part
dip Y	30 deposit instruction part
lio Y	22 load in-out register
dio Y	32 load in-out register
dzm Y	34 deposit zero in memory

Transfer Instructions

xct Y	10 execute
jmp Y	60 jump
jsp Y	62 jump and save program counter
cal Y	16 call subroutiine
jda Y	17 jump and deposit accumulator
sad Y	50 skip if accumulator and Y differ
sas Y	52 skip if accumulator and Y are the same

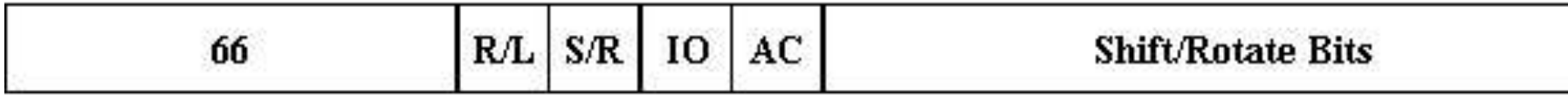


Shift/Rotate Instructions

sza	640100 skip on zero accumulator
spa	640200 skip on plus accumulator
sma	640400 skip on minus accumulator
azo	641000 skip on zero overflow
spi	642000 skip on plus io register
szs	640010-640070 skip on zero switch
szf	640001-640007 skip of zero program flag

PDP1 Instruction set

Shift/Rotate Instructions

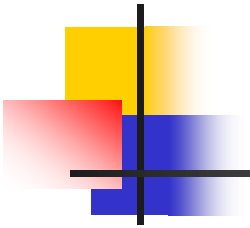


R/L
0: Right
1: Left

S/R
0: Rotate
1: Shift

IO/AC
01: Accumulator
11: IO register
11: Accumulator-IO register

Shift/Rotate Bits
Each bit set causes 1 bit shift
Allows up to 9 bit shifts

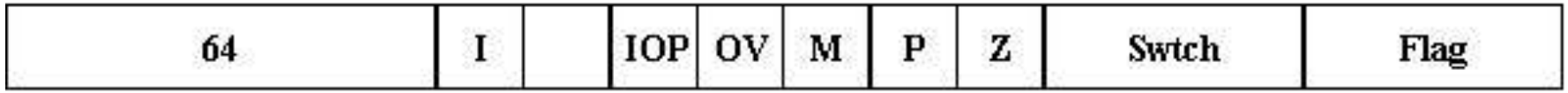


Shift/Rotate Instructions

rar	671	rotate accumulator right
ral	661	rotate accumulator left
sar	675	shift accumulator right
sal	665	shift accumulator left
rir	672	rotate io register right
ril	662	rotate io register left
sir	676	shift io register right
sil	666	shift io register left
rcr	673	rotate ac and io right
rcl	663	rotate ac and io left
scr	677	shift ac and io right
scl	667	shift ac and io left

PDP1 Instruction set

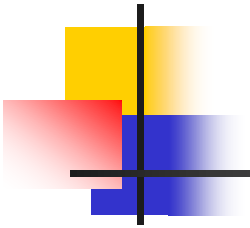
Skip Instructions



I
0: Normal
1: Invert

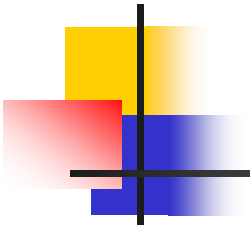
IOP Skip on plus IO
OV Skip on overflow
M Skip on minus AC
P Skip on plus AC
Z Skip on zero AC

Switch: Skip on zero swtch
Flag: Skip of zero program flag



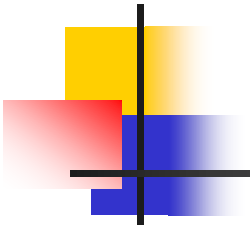
Miscelaneous Instructions

iot	72nnnn	Input-Output operations
law N	70nnnn	load accumulator with N



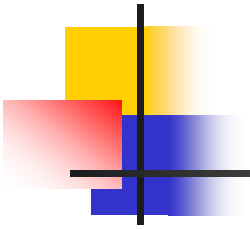
Deviations from PDP1

- Indirection bit not allowed in memory
- Memory can be incremented and decremented
- Shift / rotate one bit at a time
- IO instructions use AC, not IO register
- 2’s complement arithmetic, not 1’s complement



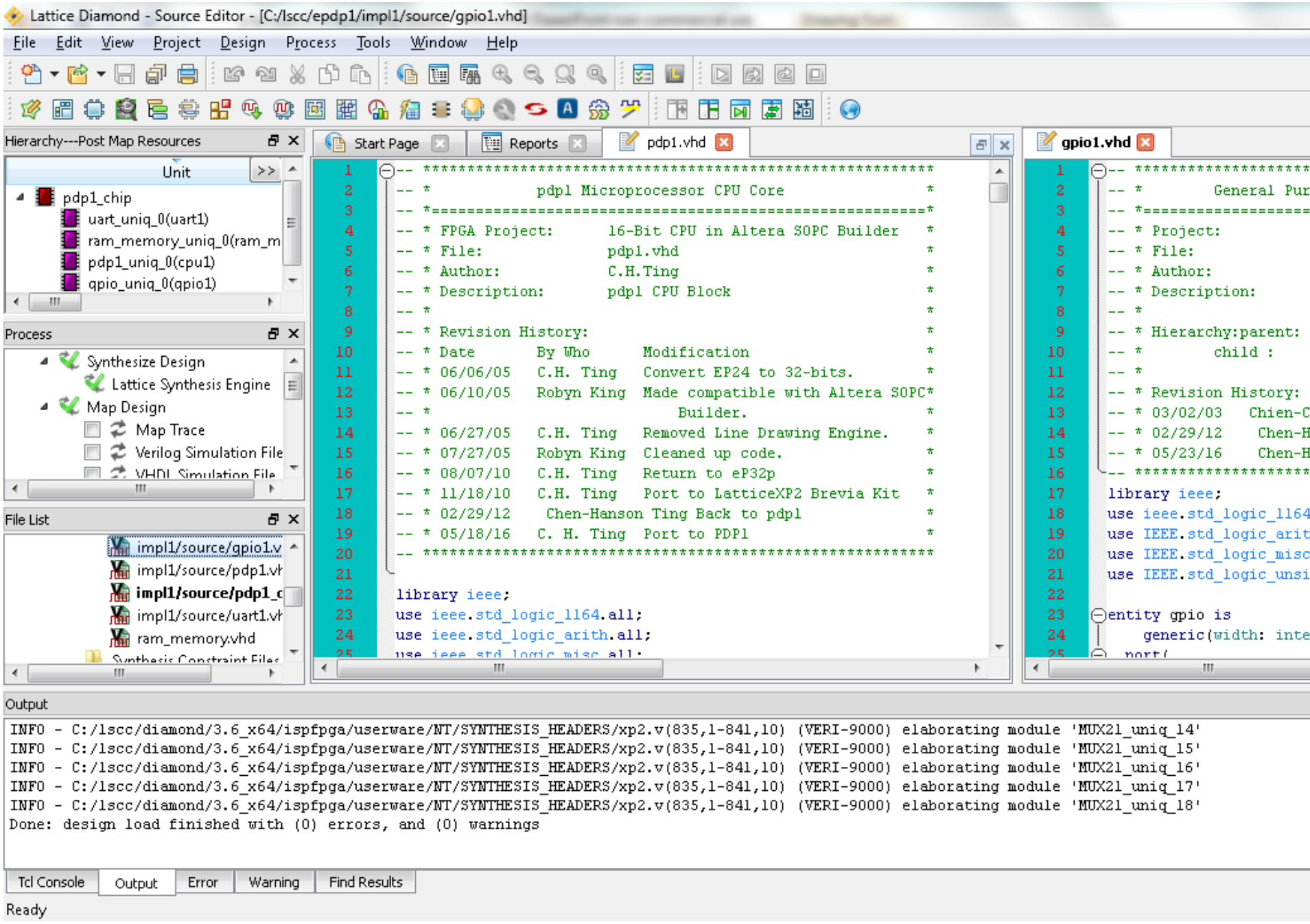
FPGA Implementation

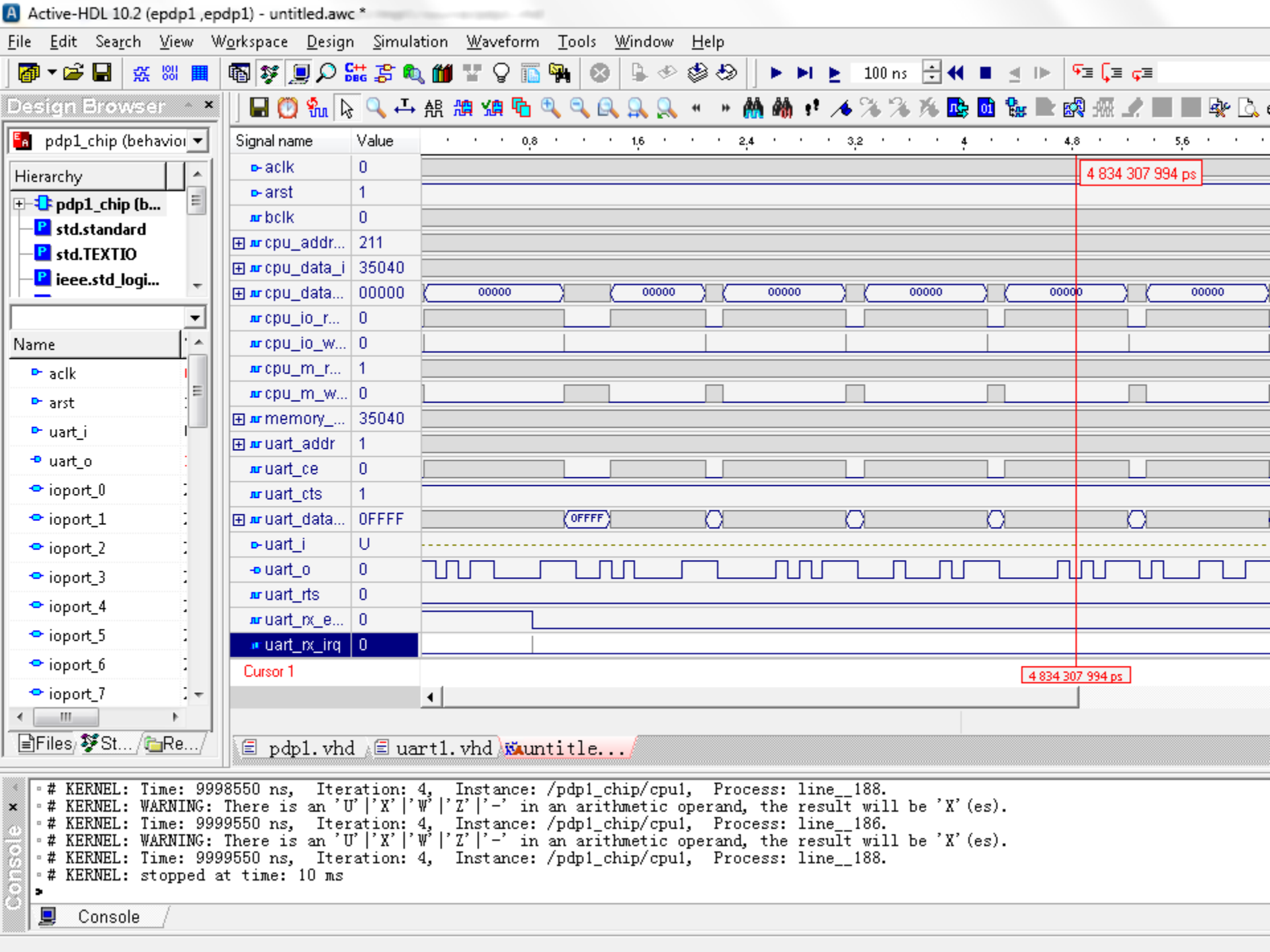
- Synthesis
- Simulation
- Programming
- Debugging

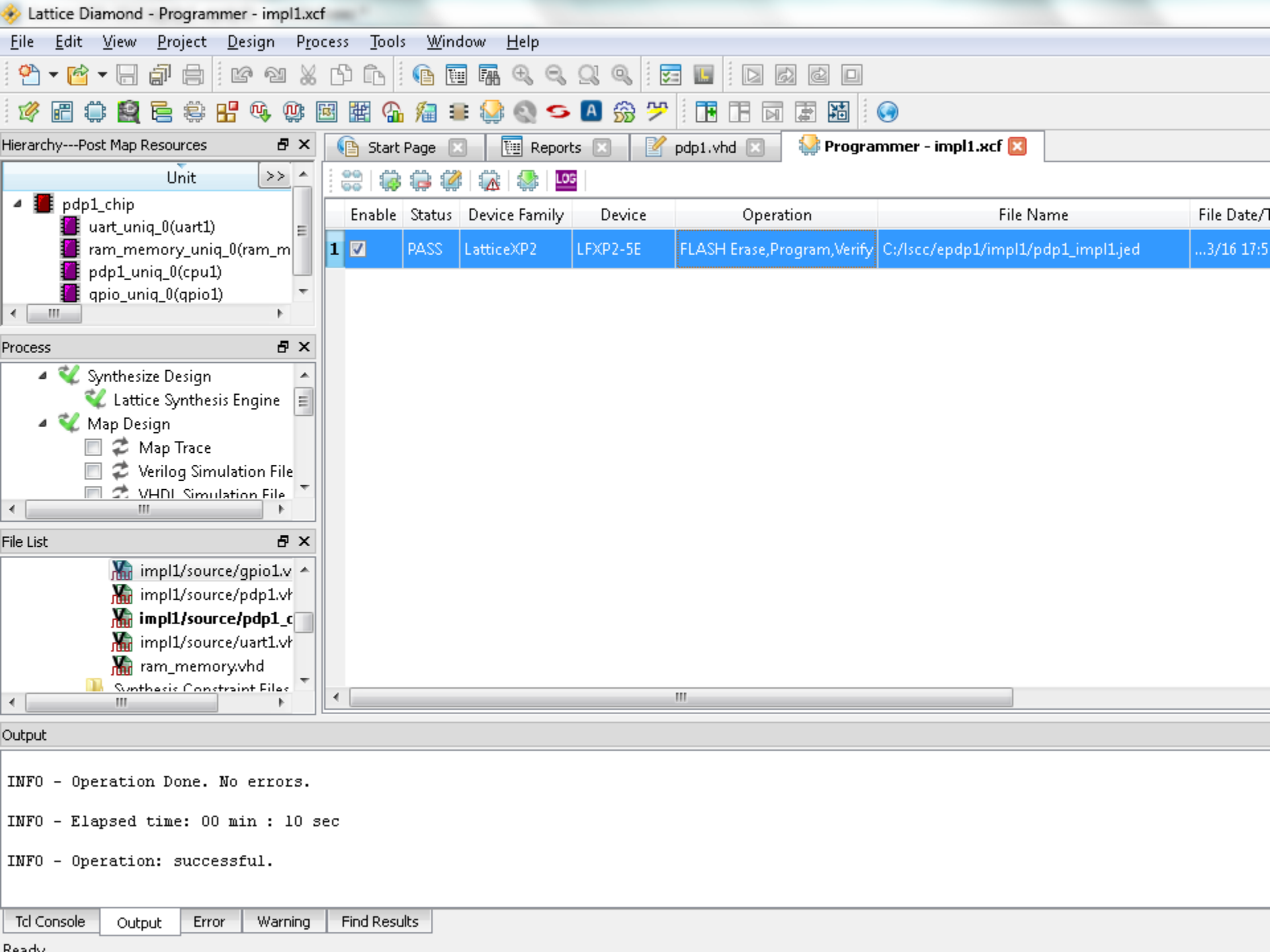


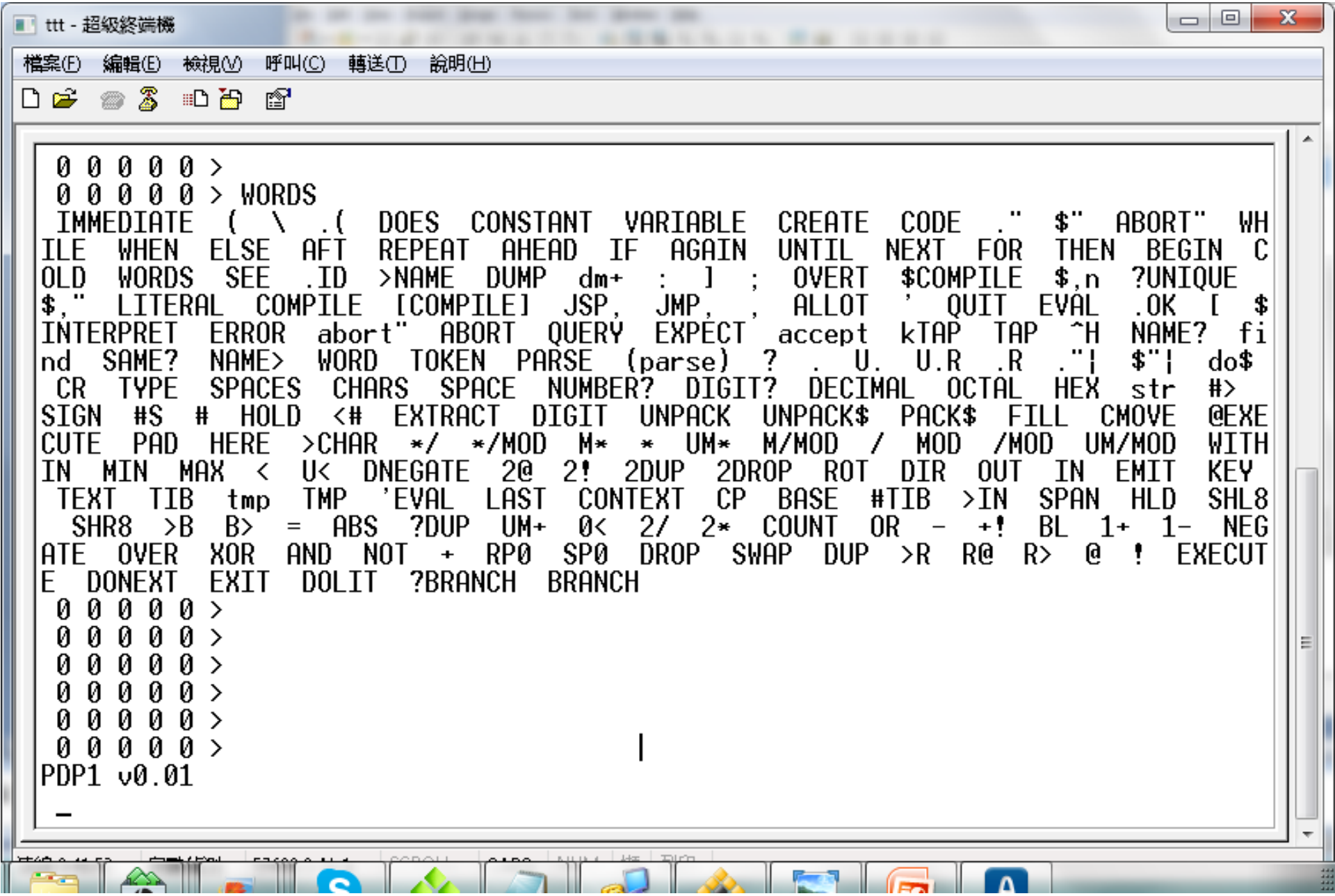
FPGA Implementation

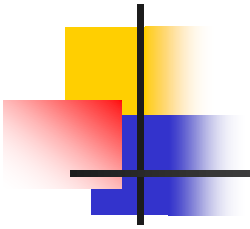
- Pdp1_chip.vhd
- Pdp1.vhd
- Uart1.vhd
- Gpio.vhd
- Pdp1.lpf


Lattice Diamond - Source Editor - [C:/lscc/epdp1/impl1/source/gpio1.vhd]
File Edit View Project Design Process Tools Window Help
Hierarchy---Post Map Resources
Unit
pdp1_chip
uart_uniq_0(uart1)
ram_memory_uniq_0(ram_m
pdp1_uniq_0(cpu1)
qpio_uniq_0(qpio1)
Process
Synthesize Design
Lattice Synthesis Engine
Map Design
Map Trace
Verilog Simulation File
File List
impl1/source/gpio1.v
impl1/source/pdp1.vh
impl1/source/pdp1_c
impl1/source/uart1.vh
ram_memory.vhd
Start Page
Reports
pdp1.vhd
-- **
-- * pdp1 Microprocessor CPU Core *
-- *==*
-- * FPGA Project: 16-Bit CPU in Altera SOPC Builder *
-- * File: pdp1.vhd *
-- * Author: C.H.Ting *
-- * Description: pdp1 CPU Block *
-- * *
-- * Revision History: *
-- * Date By Who Modification *
-- * 06/06/05 C.H. Ting Convert EP24 to 32-bits. *
-- * 06/10/05 Robyn King Made compatible with Altera SOPC*
-- * Builder. *
-- * 06/27/05 C.H. Ting Removed Line Drawing Engine. *
-- * 07/27/05 Robyn King Cleaned up code. *
-- * 08/07/10 C.H. Ting Return to eP32p *
-- * 11/18/10 C.H. Ting Port to LatticeXP2 Brevia Kit *
-- * 02/29/12 Chen-Hanson Ting Back to pdp1 *
-- * 05/18/16 C. H. Ting Port to PDP1 *
-- **
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
gpio1.vhd
-- * General Pur
-- * Project:
-- * File:
-- * Author:
-- * Description:
-- * Hierarchy:parent:
-- * child :
-- * Revision History:
-- * 03/02/03 Chien-C
-- * 02/29/12 Chen-H
-- * 05/23/16 Chen-H
library ieee;
use ieee.std_logic_1164
use IEEE.std_logic_arit
use IEEE.std_logic_misc
use IEEE.std_logic_unsi
entity gpio is
generic(width: inte
Output
INFO - C:/lscc/diamond/3.6_x64/ispfpga/userware/NT/SYNTHESIS_HEADERS/xp2.v(835,1-841,10) (VERI-9000) elaborating module 'MUX21_uniq_14'
INFO - C:/lscc/diamond/3.6_x64/ispfpga/userware/NT/SYNTHESIS_HEADERS/xp2.v(835,1-841,10) (VERI-9000) elaborating module 'MUX21_uniq_15'
INFO - C:/lscc/diamond/3.6_x64/ispfpga/userware/NT/SYNTHESIS_HEADERS/xp2.v(835,1-841,10) (VERI-9000) elaborating module 'MUX21_uniq_16'
INFO - C:/lscc/diamond/3.6_x64/ispfpga/userware/NT/SYNTHESIS_HEADERS/xp2.v(835,1-841,10) (VERI-9000) elaborating module 'MUX21_uniq_17'
INFO - C:/lscc/diamond/3.6_x64/ispfpga/userware/NT/SYNTHESIS_HEADERS/xp2.v(835,1-841,10) (VERI-9000) elaborating module 'MUX21_uniq_18'
Done: design load finished with (0) errors, and (0) warnings
Tcl Console
Output
Error
Warning
Find Results
Ready


Active-HDL 10.2 (epdp1 ,epdp1) - untitled.awc *
File Edit Search View Workspace Design Simulation Waveform Tools Window Help
100 ns
Design Browser
pdp1_chip (behavior
Hierarchy
pdp1_chip (b...
std.standard
std.TEXTIO
ieee.std_logi...
Name
aclk
arst
uart_i
uart_o
ioport_0
ioport_1
ioport_2
ioport_3
ioport_4
ioport_5
ioport_6
ioport_7
Files St... Re...
Signal name
Value
aclk 0
arst 1
bclk 0
cpu_addr... 211
cpu_data_i 35040
cpu_data... 00000
cpu_io_r... 0
cpu_io_w... 0
cpu_m_r... 1
cpu_m_w... 0
memory_... 35040
uart_addr 1
uart_ce 0
uart_cts 1
uart_data... 0FFFF
uart_i U
uart_o 0
uart_rts 0
uart_rx_e... 0
uart_rx_irq 0
0.8
1.6
2.4
3.2
4
4.8
5.6
4 834 307 994 ps
00000
0FFFF
Cursor 1
4 834 307 994 ps
pdp1.vhd
uart1.vhd
untitle...
KERNEL: Time: 9998550 ns, Iteration: 4, Instance: /pdp1_chip/cpu1, Process: line__188.
KERNEL: WARNING: There is an 'U'|'X'|'W'|'Z'|'-' in an arithmetic operand, the result will be 'X'(es).
KERNEL: Time: 9999550 ns, Iteration: 4, Instance: /pdp1_chip/cpu1, Process: line__186.
KERNEL: WARNING: There is an 'U'|'X'|'W'|'Z'|'-' in an arithmetic operand, the result will be 'X'(es).
KERNEL: Time: 9999550 ns, Iteration: 4, Instance: /pdp1_chip/cpu1, Process: line__188.
KERNEL: stopped at time: 10 ms
Console
Console

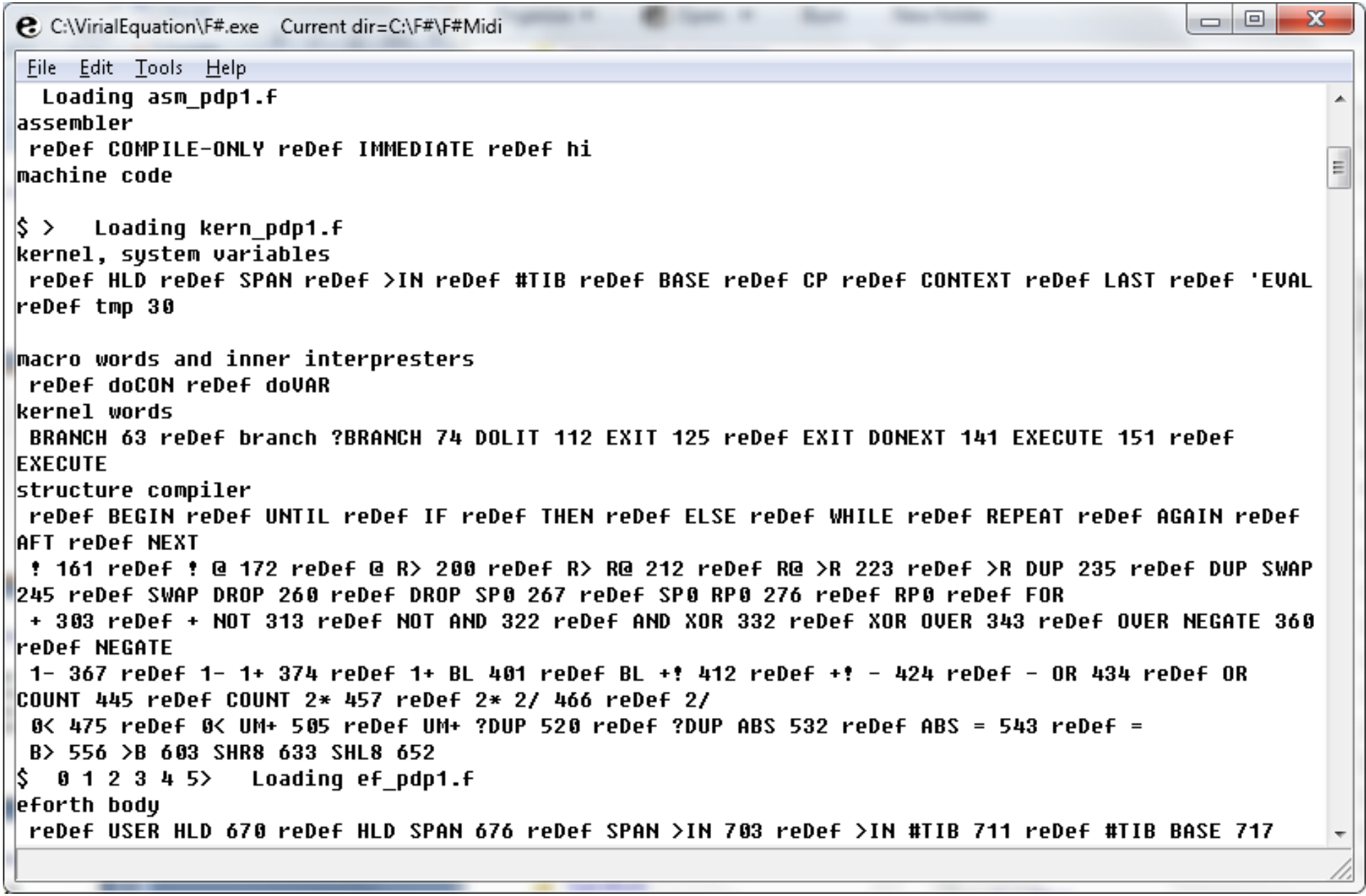

Lattice Diamond - Programmer - impl1.xcf
File Edit View Project Design Process Tools Window Help
Hierarchy---Post Map Resources
Unit
pdp1_chip
uart_uniq_0(uart1)
ram_memory_uniq_0(ram_m
pdp1_uniq_0(cpu1)
qpio_uniq_0(qpio1)
Process
Synthesize Design
Lattice Synthesis Engine
Map Design
Map Trace
Verilog Simulation File
File List
impl1/source/gpio1.v
impl1/source/pdp1.vh
impl1/source/pdp1_c
impl1/source/uart1.vh
ram_memory.vhd
Start Page
Reports
pdp1.vhd
Programmer - impl1.xcf
Enable
Status
Device Family
Device
Operation
File Name
File Date/T
1
PASS
LatticeXP2
LFXP2-5E
FLASH Erase,Program,Verify
C:/lscc/epdp1/impl1/pdp1_impl1.jed
...3/16 17:5
Output
INFO - Operation Done. No errors.
INFO - Elapsed time: 00 min : 10 sec
INFO - Operation: successful.
Tcl Console
Output
Error
Warning
Find Results
Ready

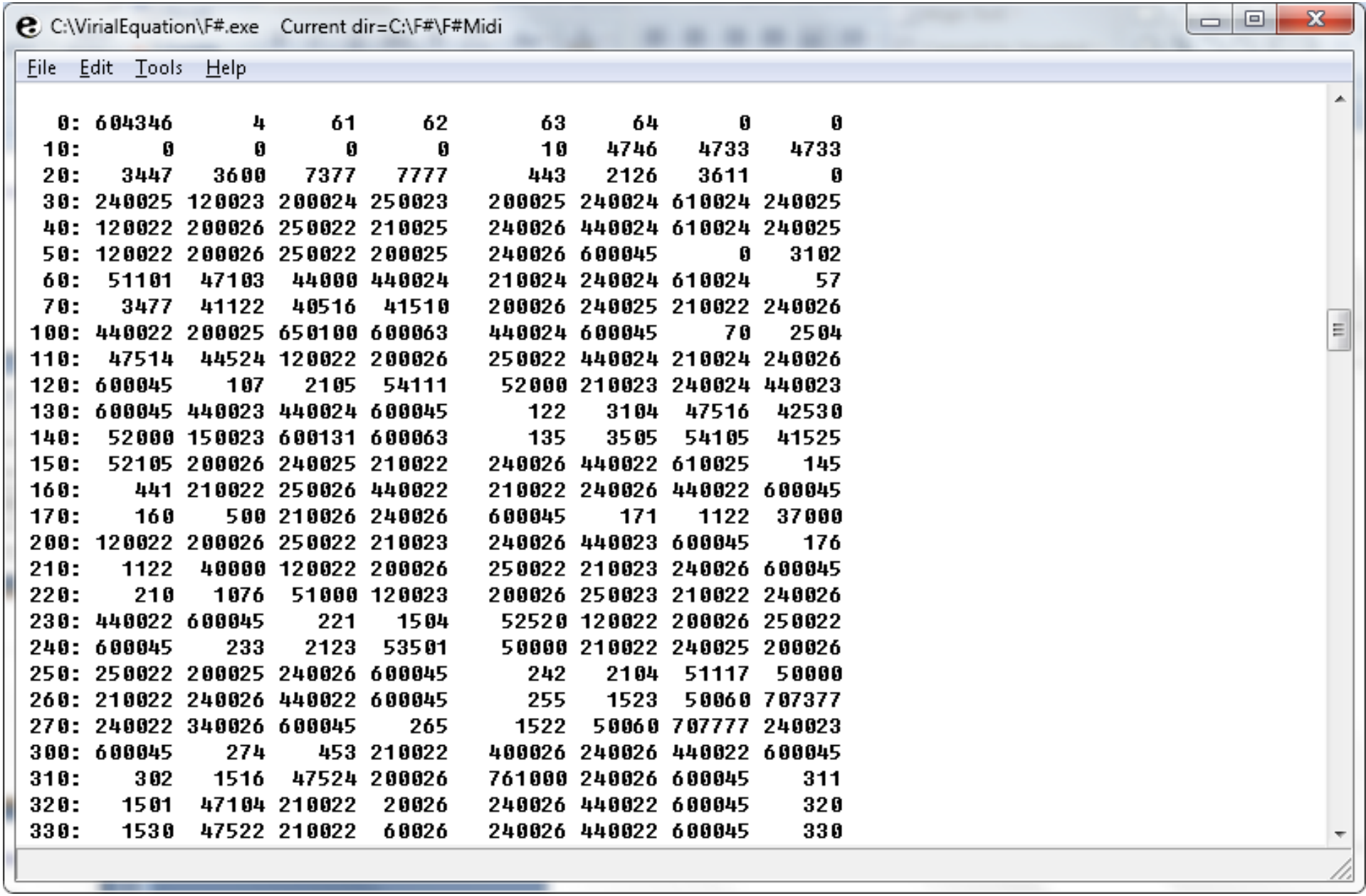

ttt - 超級終端機
檔案(F) 編輯(E) 檢視(V) 呼叫(C) 轉送(T) 說明(H)
0 0 0 0 0 >
0 0 0 0 0 > WORDS
IMMEDIATE (\ .(DOES CONSTANT VARIABLE CREATE CODE ." $" ABORT" WH
ILE WHEN ELSE AFT REPEAT AHEAD IF AGAIN UNTIL NEXT FOR THEN BEGIN C
OLD WORDS SEE .ID >NAME DUMP dm+ :] ; OVERT $COMPILE $,n ?UNIQUE
$," LITERAL COMPILE [COMPILE] JSP, JMP, , ALLOT ' QUIT EVAL .OK [$
INTERPRET ERROR abort" ABORT QUERY EXPECT accept kTAP TAP ^H NAME? fi
nd SAME? NAME> WORD TOKEN PARSE (parse) ? . U. U.R .R .¦ $"¦ do$
CR TYPE SPACES CHARS SPACE NUMBER? DIGIT? DECIMAL OCTAL HEX str #>
SIGN #S # HOLD <# EXTRACT DIGIT UNPACK UNPACK$ PACK$ FILL CMOVE @EXE
CUTE PAD HERE >CHAR */ */MOD M* * UM* M/MOD / MOD /MOD UM/MOD WITH
IN MIN MAX < U< DNEGATE 2@ 2! 2DUP 2DROP ROT DIR OUT IN EMIT KEY
TEXT TIB tmp TMP 'EVAL LAST CONTEXT CP BASE #TIB >IN SPAN HLD SHL8
SHR8 >B B> = ABS ?DUP UM+ 0< 2/ 2* COUNT OR - +! BL 1+ 1- NEG
ATE OVER XOR AND NOT + RP0 SP0 DROP SWAP DUP >R R@ R> @ ! EXECUT
E DONEXT EXIT DOLIT ?BRANCH BRANCH
0 0 0 0 0 >
0 0 0 0 0 >
0 0 0 0 0 >
0 0 0 0 0 >
0 0 0 0 0 >
0 0 0 0 0 >
PDP1 v0.01

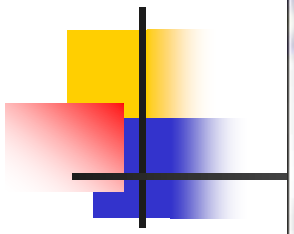


eForth Implementation

- Pdp1.fex
- Meta_pdp1.f
- Asm_pdp1.f
- Kernel_pdp1.f
- Ef_pdp1.f
- Pdp1.mem


C:\VirialEquation\F#.exe Current dir=C:\F#\F#Midi
File Edit Tools Help
Loading asm_pdp1.f
assembler
reDef COMPILE-ONLY reDef IMMEDIATE reDef hi
machine code
$ > Loading kern_pdp1.f
kernel, system variables
reDef HLD reDef SPAN reDef >IN reDef #TIB reDef BASE reDef CP reDef CONTEXT reDef LAST reDef 'EVAL
reDef tmp 30
macro words and inner interpresters
reDef doCON reDef doVAR
kernel words
BRANCH 63 reDef branch ?BRANCH 74 DOLIT 112 EXIT 125 reDef EXIT DONEXT 141 EXECUTE 151 reDef
EXECUTE
structure compiler
reDef BEGIN reDef UNTIL reDef IF reDef THEN reDef ELSE reDef WHILE reDef REPEAT reDef AGAIN reDef
AFT reDef NEXT
! 161 reDef ! @ 172 reDef @ R> 200 reDef R> R@ 212 reDef R@ >R 223 reDef >R DUP 235 reDef DUP SWAP
245 reDef SWAP DROP 260 reDef DROP SP0 267 reDef SP0 RP0 276 reDef RP0 reDef FOR
+ 303 reDef + NOT 313 reDef NOT AND 322 reDef AND XOR 332 reDef XOR OVER 343 reDef OVER NEGATE 360
reDef NEGATE
1- 367 reDef 1- 1+ 374 reDef 1+ BL 401 reDef BL +! 412 reDef +! - 424 reDef - OR 434 reDef OR
COUNT 445 reDef COUNT 2* 457 reDef 2* 2/ 466 reDef 2/
0< 475 reDef 0< UM+ 505 reDef UM+ ?DUP 520 reDef ?DUP ABS 532 reDef ABS = 543 reDef =
B> 556 >B 603 SHR8 633 SHL8 652
$ 0 1 2 3 4 5> Loading ef_pdp1.f
eforth body
reDef USER HLD 670 reDef HLD SPAN 676 reDef SPAN >IN 703 reDef >IN #TIB 711 reDef #TIB BASE 717


C:\VirialEquation\F#.exe Current dir=C:\F#\F#Midi
File Edit Tools Help
0: 604346 4 61 62 63 64 0 0
10: 0 0 0 0 10 4746 4733 4733
20: 3447 3600 7377 7777 443 2126 3611 0
30: 240025 120023 200024 250023 200025 240024 610024 240025
40: 120022 200026 250022 210025 240026 440024 610024 240025
50: 120022 200026 250022 200025 240026 600045 0 3102
60: 51101 47103 44000 440024 210024 240024 610024 57
70: 3477 41122 40516 41510 200026 240025 210022 240026
100: 440022 200025 650100 600063 440024 600045 70 2504
110: 47514 44524 120022 200026 250022 440024 210024 240026
120: 600045 107 2105 54111 52000 210023 240024 440023
130: 600045 440023 440024 600045 122 3104 47516 42530
140: 52000 150023 600131 600063 135 3505 54105 41525
150: 52105 200026 240025 210022 240026 440022 610025 145
160: 441 210022 250026 440022 210022 240026 440022 600045
170: 160 500 210026 240026 600045 171 1122 37000
200: 120022 200026 250022 210023 240026 440023 600045 176
210: 1122 40000 120022 200026 250022 210023 240026 600045
220: 210 1076 51000 120023 200026 250023 210022 240026
230: 440022 600045 221 1504 52520 120022 200026 250022
240: 600045 233 2123 53501 50000 210022 240025 200026
250: 250022 200025 240026 600045 242 2104 51117 50000
260: 210022 240026 440022 600045 255 1523 50060 707377
270: 240022 340026 600045 265 1522 50060 707777 240023
300: 600045 274 453 210022 400026 240026 440022 600045
310: 302 1516 47524 200026 761000 240026 600045 311
320: 1501 47104 210022 20026 240026 440022 600045 320
330: 1530 47522 210022 60026 240026 440022 600045 330

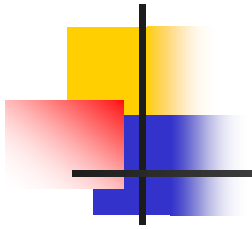


pdp1.mem - Notepad

File Edit Format View Help

```
#Format=AddrHex
#Depth=4096
#Width=18
#AddrRadix=3
#DataRadix=3
0:308E6
1:4
2:31
3:32
4:33
5:34
C:8
D:9E6
E:9DB
F:9DB
10:727
11:780
12:EFF
13:FFF
14:123
15:456
16:789
18:14015
19:A013
1A:10014
1B:15013
1C:10015
1D:14014
1E:31014
1F:14015
20:A012
21:10016
22:15012
23:11015
24:14016
25:24014
```

Forth Virtual Machine on PDP1



- Hardware registers: AC, IO
- FVM registers: ac, ip, sp, rp, tos, tmp
- Inner interpreters:

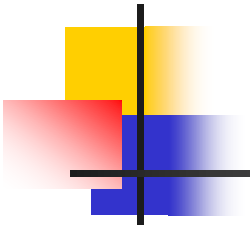
  next: ip idx ip () jmp

  doLIST

  EXIT

  doVAR

  doCON

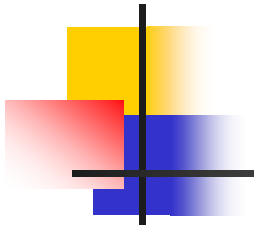


eForth Inner Interpreter

```
22 CONSTANT sp
23 CONSTANT rp
24 CONSTANT ip
25 CONSTANT ac
26 CONSTANT tos
27 CONSTANT tmp
```

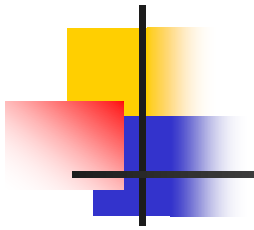
eForth Inner Interpreter

```
next:     ip idx ip () jmp
doLIST:   ac dac rp ddx ip lac rp () dac
          ac lac ip dac ip () jmp
EXIT:     rp () lac ip dac rp idx next
doCON:    ac dac pushs ac () lac tos dac next
doVAR:    ac dac pushs ac lac tos dac next
```

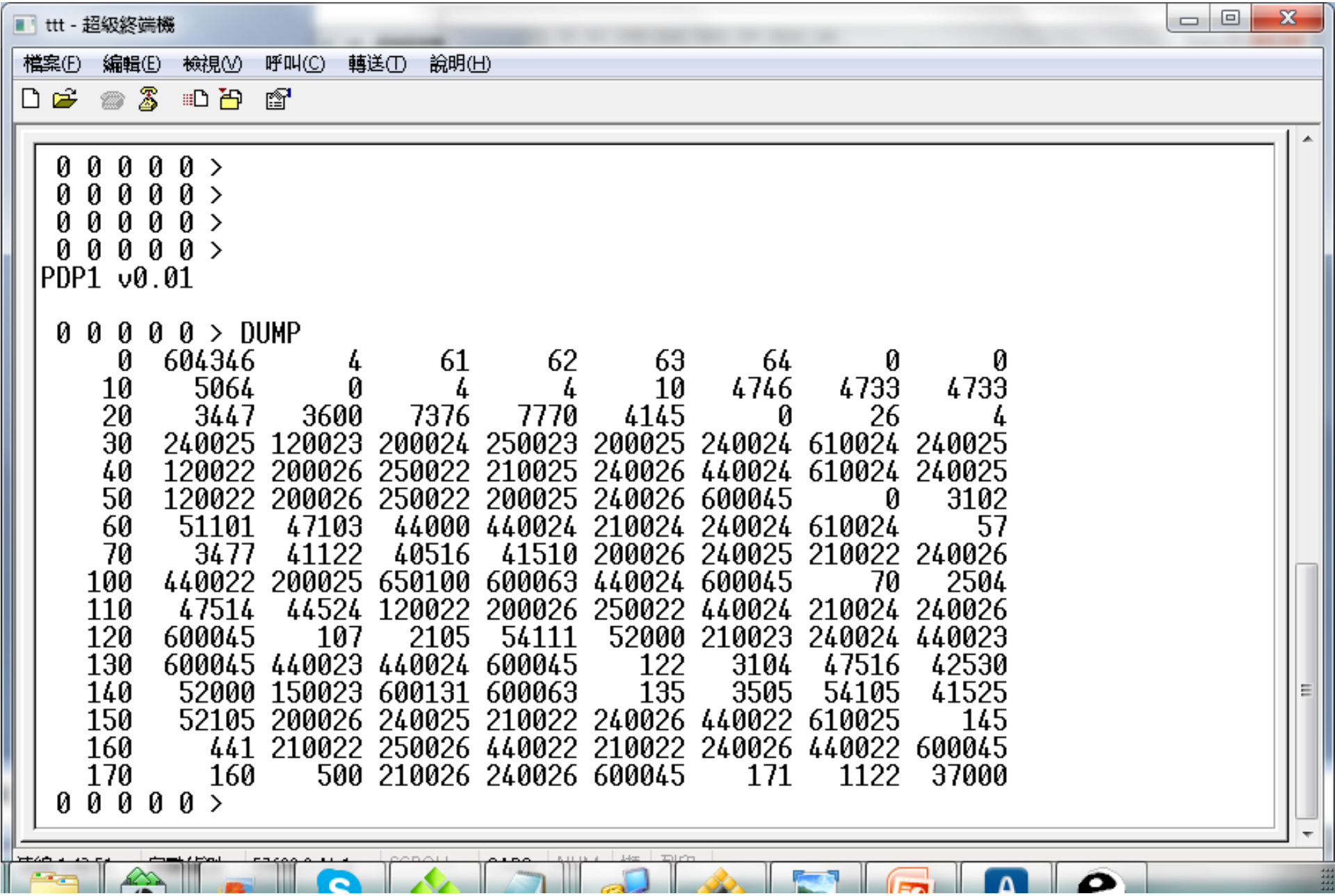


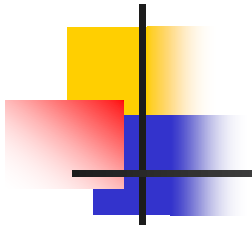
WORDS

```
:: WORDS ( -- )
  CR CONTEXT
  BEGIN @ ?DUP
  WHILE DUP SPACE .ID
    1-
  REPEAT ;;
```



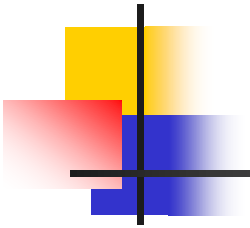
```
2527 47522 42123  WORDS
620030            doLIST
602457            CR
600733            CONTEXT
                  BEGIN
600172            @
600520            ?DUP
600074 4341       WHILE
600235            DUP
602374            SPACE
604233            .ID
600367            1-
600063 4327       REPEAT
600125            EXIT
```


ttt - 超級終端機
檔案(F) 編輯(E) 檢視(V) 呼叫(C) 轉送(T) 說明(H)
0 0 0 0 0 >
0 0 0 0 0 >
0 0 0 0 0 >
0 0 0 0 0 >
PDP1 v0.01
0 0 0 0 0 > DUMP
0 604346 4 61 62 63 64 0 0
10 5064 0 4 4 10 4746 4733 4733
20 3447 3600 7376 7770 4145 0 26 4
30 240025 120023 200024 250023 200025 240024 610024 240025
40 120022 200026 250022 210025 240026 440024 610024 240025
50 120022 200026 250022 200025 240026 600045 0 3102
60 51101 47103 44000 440024 210024 240024 610024 57
70 3477 41122 40516 41510 200026 240025 210022 240026
100 440022 200025 650100 600063 440024 600045 70 2504
110 47514 44524 120022 200026 250022 440024 210024 240026
120 600045 107 2105 54111 52000 210023 240024 440023
130 600045 440023 440024 600045 122 3104 47516 42530
140 52000 150023 600131 600063 135 3505 54105 41525
150 52105 200026 240025 210022 240026 440022 610025 145
160 441 210022 250026 440022 210022 240026 440022 600045
170 160 500 210026 240026 600045 171 1122 37000
0 0 0 0 0 >



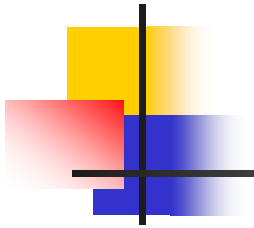
Demonstrations

```
: TEST1 1 2 3 4 5 ;
: TEST2 CR ." HELLO, WORLD!" ;
: TEST3 IF 1 ELSE 2 THEN . ;
: TEST4 10 FOR R@ . NEXT ;
: TEST5 10 BEGIN DUP . DUP
          WHILE 1- REPEAT DROP ;
```



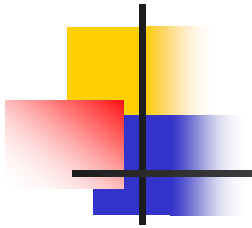
Demonstrations

HEX	from octal
0 OUT	turn on all LED’s
FE OUT	turn on LED-1
IN	read switches
IN	read switches
IN	read switches

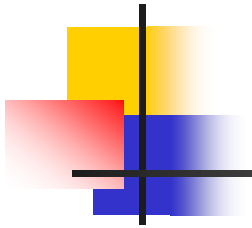


Last Words

The easiest way
to design a CPU
is the eForth way.



Questions?



Thank you very much.